Fat and Thin Adaptive HMM Filters for Vision Based Detection of Moving Targets

Alexander Wainwright, Jason J. Ford and John Lai*

Abstract

Computer vision is an attractive solution for uninhabited aerial vehicle (UAV) collision avoidance, due to the low weight, size and power requirements of hardware. A two-stage paradigm has emerged in the literature for detection and tracking of dim targets in images, comprising of spatial preprocessing, followed by temporal filtering. In this paper, we investigate a hidden Markov model (HMM) based temporal filtering approach. Specifically, we propose an adaptive HMM filter, in which the variance of model parameters is refined as the quality of the target estimate improves. Filters with high variance (fat filters) are used for target acquisition, and filters with low variance (thin filters) are used for target tracking. The adaptive filter is tested in simulation and with real data (video of a collision-course aircraft). Our test results demonstrate that our adaptive filtering approach has improved tracking performance, and provides an estimate of target heading not present in previous HMM filtering approaches.

1 INTRODUCTION

There is a general consensus among regulators and aviation authorities that uninhabited aerial vehicles (UAVs) cannot be safely integrated into civil airspace unless a sense-and-avoid capability with an equivalent performance to human pilot see-and-avoid can be demonstrated [DeGarmo, 2004].

Computer vision is an attractive approach to addressing the sensing aspect of the sense-and-avoid problem, due to the small size, low weight, and low power requirements of sensor hardware [Lai *et al.*, 2010]. When computer vision detection and tracking techniques are applied in high signal to noise ratio (SNR) environments, simple thresholding on each frame can be effective; however, when the SNR is low, thresholding approaches can result in much of the signal information being lost. Hence, in low SNR situations, approaches in which the target is tracked over a number of frames before a detection is declared are preferred. Such temporal filtering approaches are referred to as track-before-detect (TBD) [Barniv, 1985; Tonissen and Evans, 1996], and include Viterbi based techniques [Barniv, 1985; Barniv and Kella, 1987; Tonissen and Evans, 1996; Gandhi *et al.*, 2006; Carnie *et al.*, 2006; Lai *et al.*, 2010], and Bayesian hidden Markov model (HMM) filtering [Bruno and Moura, 2001; Bruno, 2004; Lai *et al.*, 2008a; Lai and Ford, 2010; Lai *et al.*, 2010]. Recent comparison studies of several TBD algorithms has shown the Bayesian HMM filter to be close to state-of-the-art [Lai *et al.*, 2010; Davey *et al.*, 2008; Lai *et al.*, 2008b]. In [Lai *et al.*, 2008a], HMM filtering concepts are extended by considering a bank of four HMM filters, where each filter is looking for targets with different headings in the image. This study showed that a HMM filter bank is more effective than a single HMM filter.

A systematic design approach for designing HMM filter banks using the concept of relative entropy rates was introduced in [Lai and Ford, 2010]. The effectiveness of the HMM filter for target tracking was demonstrated in [Lai *et al.*, 2010] with flight testing, where HMM filters, in combination with morphological preprocessing, could detect a target aircraft 8–10s before collision.

We extend the filter bank concept presented in [Lai *et al.*, 2008a], by proposing an adaptive HMM filter where model designs are based on expected target motion. This adaptive filter has improved detection performance and generates an estimate of target heading angle in the image plane. The main contribution of this paper is the investigation of the benefits of an adaptive filter design

*The authors are with the Australian Research Centre for Aerospace Automation, School of Engineering Systems, Faculty of Build Environment and Engineering, Queensland University of Technology, Brisbane, QLD, 4001 Australia alexander.wainwright@qut.edu.au, j2ford@qut.edu.au, js.lai@qut.edu.au

where the variance of the filter model parameters is refined as the quality of the target estimate is improved; i.e. the expected variance of the target heading is adjusted with changes in certainty of the target track. A large variance (or "fat") filter is required when there is high uncertainty of the target motion, such as during the target acquisition phase. On the other hand a small variance (or "thin") filter can be introduced once the filter is tracking the target.

This paper is structured as follows. In Section 2 the target model, problem statement and solution structure are introduced. In Section 3, the morphological preprocessing stage is presented. In Section 4 the HMM approximation, HMM filter, and filter design methods are introduced. In Section 5, we introduce the "fat" and "thin" patch concepts, and an adaptive HMM filter. In Section 6, we discuss the implementation of the algorithm on a graphics processing unit (GPU). In Section 7, simulations are carried out to investigate the properties and performance of the thin patches. Finally, in Section 8, some conclusions are drawn.

2 Target dynamics and problem formulation

In this section we introduce our 2D target dynamics and measurement model, and then present the two-stage filtering paradigm.

2.1 Target dynamics

We are concerned with detecting and tracking a small, sub-pixel or spot target in a sequence of 2D images of size $m \times n$ (discrete time and discrete space), with location in the image plane denoted $x = [i, j] \in \mathbb{R}^2$, where i is the location in the horizontal direction and j is the location in the vertical direction. The target motion is modelled as [Lai and Ford, 2010]:

$$x_{k+1} = x_k + \begin{bmatrix} v^i \\ v^j \end{bmatrix}, \; x_0 \in \mathbb{R}^2, \tag{1}$$

where $k = 1, 2, 3 \ldots$ is the discrete time step, and $(v^i, v^j) \in \mathbb{R}^2$ is the target velocity in pixels per frame.

The target state, x_k, is observed indirectly by the image measurement model:

$$y_k = c(x_k) + w_k, \tag{2}$$

where $y \in \mathbb{R}^{m \times n}$ is the observed $m \times n$ image, $w_k \in \mathbb{R}^{m \times n}$ is the measurement noise with density $\varphi_w(.)$ and $c : \mathbb{R}^2 \to \mathbb{R}^{m \times n}$.

In the following it is assumed all noise processes are mutually independent.

2.2 Problem statement and solution structure

The problem considered in this paper is to reliably track a target described by (1) and (2), in an image with a sky background, for an airborne collision avoidance application.

In the aircraft detection and tracking problem, target aircraft are small, pin-like features in the image, whereas clutter (such as clouds) is often present as larger structures. Therefore, spatial filtering can be useful in separating spot targets from other non-target features in the image. However, in low SNR environments, thresholding performed on spatially filtered images is often inadequate for reliably detecting targets, so temporal filters, in which the target is tracked over a number of frames before a detection is declared, are also used. Hence we have a two-stage processing paradigm:

1. spatial filtering, then
2. temporal filtering.

We expect that the majority of clutter features will be removed by a combination of the above two stages. Here, we use a morphological filter as the spatial filtering stage which will be discussed in Section 3; however, the main contribution of this paper is related to novel solutions for temporal filtering which will be discussed in Section 4 and Section 5.

3 Morphological preprocessing

In this paper, we assume that all raw measurements undergo morphological filtering (the spatial filtering stage) before being processed by the temporal filter. The morphological filter we use is described below.

The fundamental morphological operations are the erosion, $\ominus$, and dilation, $\oplus$, which are operations with two operands: an image, Y, and a structuring element, S, see [Dougherty and Lotufo, 2003] for details. Combinations of these operations can be used to form more sophisticated morphological operations, such as the morphological opening operation, $\circ$, defined as [Dougherty and Lotufo, 2003; Lai *et al.*, 2008b]:

$$Y \circ S = (Y \ominus S) \oplus S, \tag{3}$$

and the morphological closing operation, $\bullet$, defined as:

$$Y \bullet S = (Y \oplus S) \ominus S. \tag{4}$$

The opening operation can be thought of as darkening locally bright regions, and the closing operation as brightening locally dark regions [Lai *et al.*, 2008b]. Using this information, a useful operation known as a "bottom-hat" filter can be defined [Gonzalez *et al.*, 2004]:

$$BH(Y, S) = (Y \bullet S) - Y. \tag{5}$$

This bottom-hat filter will emphasise any locally dark regions that are smaller than the structuring element [Lai *et al.*, 2008b]. A study has shown that in the airborne collision avoidance application, targets are usually darker than the background noise [Geyer *et al.*, 2009]. For this reason, all images undergo preprocessing with a bottom-hat morphological filter. However, we highlight that in situations where the target is brighter than the background, the dual to the bottom-hat filter, known as a "top-hat" filter could be used.

A 5×5 structuring element is used. The dimensions of the structuring element are related to the target size.

4 HMM FILTERING AND DESIGN

In this section we discuss the HMM filter used for target detection and tracking (the temporal filtering stage), and introduce the fat and thin patches used in the HMM filter. In Subsection 4.1 we introduce the HMM approximation of the target dynamics. Subsection 4.2 describes the HMM filter used for target detection and tracking, and Subsection 4.3 defines the detection statistic used in the HMM filter. In Subsection 4.4 we introduce the concept of transition patches to describe target motion. In Subsection 4.5 the properties of the target dynamics, presented in terms of the transition patches are discussed. In Subsection 4.6 we describe how we use these target dynamic properties in the design of fat and thin filters.

4.1 Hidden Markov model

In this section we introduce an HMM approximation of the target dynamics in (1)-(2), also see [Lai and Ford, 2010].

We assume a small, sub-pixel target on a discrete, two dimensional, image grid with $N = m \times n$ pixels. Let $e_i = [0, \ldots, 0, 1, 0, \ldots, 0]'$ denote a $N \times 1$ indicator vector with 1 in the ith element and 0 elsewhere. The state of the target at time k is denoted X_k, with $X_k \in \{e_1, e_2, \ldots, e_N\}$, where each indicator vector maps to a unique pixel in the image, i.e., pixel coordinates (i, j) map to the rth element of the indicator vector as follows:

$$r = (i-1) \times n + j, \tag{6}$$

for $i \in \{1, 2, \ldots, m\}$, $j \in \{1, 2, \ldots, n\}$ giving $r \in \{1, 2, \ldots, N\}$.

The temporal behaviour of the chain is described by the transition probability matrix, $A \in \mathbb{R}^{N \times N}$, where $A_{i,j} = P(X_{k+1} = e_i | X_k = e_j)$.

The measurement probability matrix is denoted $B(y_k) \in \mathbb{R}^{N \times N}$, and is defined as:

$$B(y_k)_{i,j} = \begin{cases} P(y_k | X_k = e_i) & \text{if } i = j \\ 0 & \text{elsewhere} \end{cases} \tag{7}$$

4.2 The HMM filter

The conditional mean estimate of X_k given measurements up to time k is denoted $\hat{X}_k$. The HMM filter recursively calculates $\hat{X}_k$ using the following equation [Lai and Ford, 2010]:

$$\hat{X}_k = N_k B(y_k) A \hat{X}_{k-1}, \tag{8}$$

where N_k is a scaling factor given by,

$$N_k = \frac{1}{\langle B(y_k) A \hat{X}_{k-1}, \mathbb{1}_N \rangle}, \tag{9}$$

where $\langle ., . \rangle$ denotes the scalar dot product, and $\mathbb{1}_N = [1, 1, \ldots, 1]$ is the $1 \times N$ vector of all ones.

Let $\hat{X}_0 = p_0$, where p_0 is an $N \times 1$ vector of initial probabilities. In the absence of any *a priori* information of the state, each element of p_0 is set to $1/N$.

4.3 Detection declaration

The filter (8) provides an estimate of the target position in the image, but we are also interested in determining whether or not a target is present. The scale factor, N_k is easily calculated from (9). N_k is related to the probability of measurements up to time k assuming a target is present [Lai *et al.*, 2010]:

$$P(y_1, y_2, ..., y_k) = \prod_{l=1}^{k} \frac{1}{N_l}, \tag{10}$$

and can therefore be used to test the target present hypothesis given a measurement sequence. We are more interested in the likelihood of the recent part of the observation sequence, rather than the whole observation sequence, so we use an exponential moving average of $\log \frac{1}{N_k}$. Using the sum of logs rather than the product of N_k values has the added benefit of preventing floating point underflow. Hence, our detection metric, denoted τ_k, is given by:

$$\tau_k = \alpha \log \left(\frac{1}{N_k} \right) + (1 - \alpha) \tau_{k-1}, \tag{11}$$

where α is the weighting coefficient for the moving average. When τ_k crosses a detection threshold, denoted γ, a detection is declared.

4.4 Transition patches

Studies have shown that collision-course aircraft appear as slow moving features on the image plane [Limitations of the See-and-Avoid Principle, 1991]. This motivates us to constrain the state dynamics in our HMM approximation in a manner such that only self-transitions and transitions to nearby pixels in consecutive frames are permitted (i.e., HMMs corresponding to targets with slow

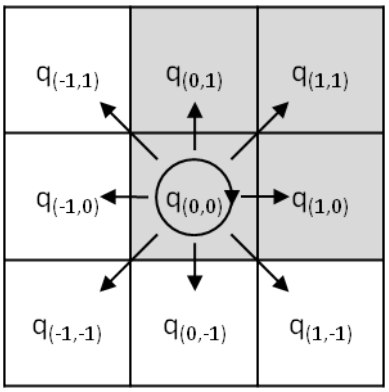

Figure 1: Target transitions in a patch, q. Each cell represents a pixel. The value $q_{(i,j)}$ represents the probability of transitioning i pixels horizontally, and j pixels vertically between frames. The darkened cells indicate elements which would be non-zero in an example quadrant patch.

motion in the image plane). Figure 1 illustrates motion on the image that is constrained to eight adjacent pixels. We will refer to non-zero transition probabilities in Figure 1 as a "transition patch." If we assume that target dynamics are independent of pixel location, then the transition patch representation of state dynamics provides a convenient descriptive shorthand for the full $N \times N$ transition probability matrix, A. Later in the paper, we will consider the design of A in terms of design of transition patches.

4.5 Properties of dynamics

The transition probability matrix, and hence the transition patch, represents the expected motion of the target. In this section we present some properties of these dynamics. An example patch, q, is shown in Figure 1, with elements $q_{(i,j)}$ representing the probability of the target moving i pixels in the horizontal direction, and j pixels in the vertical direction, in a single time step. Due to the probabilistic interpretation (each patch represents all possible transitions), we can write

$$\sum_{i,j} q_{(i,j)} = 1. \tag{12}$$

One way we can link patch probabilities to the more familiar descriptors of target motion like speed and heading angle is via the following relationships. These expressions allow us to gain insight into the dynamics patch probabilities represent. In the next section we will consider these equations in reverse and consider design of patch probabilities to meet particular expected dynamics.

From basic principles, we can calculate the expected speed (in pixels per frame) of the motion described by the patch as follows:

$$\bar{v}(q) = E[v] = \sum_{i=-1}^{1} \sum_{j=-1}^{1} q_{(i,j)} \times \sqrt{i^2 + j^2}. \tag{13}$$

The expected heading of the motion described by the patch is given by

$$\bar{\theta}(q) = E[\theta] = \sum_{i=-1}^{1} \sum_{j=-1}^{1} q_{(i,j)} \times \arctan(j/i). \tag{14}$$

Similarly, we can calculate heading variance of the motion described by the patch as follows,

$$\bar{\sigma}_\theta^2(q) = E[(\theta - \bar{\theta})^2] = \sum_{i=-1}^{1} \sum_{j=-1}^{1} q_{(i,j)} \times (\arctan(j/i) - E[\theta])^2. \tag{15}$$

4.6 Design of transition patches

Using equations (13)-(15), patch probabilities can be designed for desired target statistics expressed in terms of mean speed, mean heading, and heading variance. If the quadrant of the target's heading is known or if we are only interested in one quadrant, then the transition probabilities can be restricted to this quadrant. For a 3×3 patch, restricting target motion to one quadrant results in four transition probabilities to be determined (illustrated by the grey cells in Figure 1).

Since $\sum_{i,j} q_{(i,j)} = 1$, only three transition probability variables need to be selected for a 3×3 patch. Analytical solutions for these transition probabilities based on desired $\bar{v}$, $\bar{\theta}$ and $\bar{\sigma}_\theta^2$ are difficult to find. Here, we employ numerical methods to find a solution.

In order to find a patch in the $[0 - \frac{\pi}{2}]$ quadrant with the desired dynamics, we select patch probabilities according to the following cost function:

$$J(q) = (\bar{v}_d - \bar{v}(q))^2 + (\bar{\theta}_d - \bar{\theta}(q))^2 + (\bar{\sigma}_{\theta,d}^2 - \bar{\sigma}_\theta^2(q))^2, \tag{16}$$

where $\bar{v}_d$, $\bar{\theta}_d$ and $\bar{\sigma}_{\theta,d}^2$ represent the desired $\bar{v}$, desired $\bar{\theta}$ and desired $\bar{\sigma}_\theta^2$, respectively. We find a q which minimises $J(q)$, with the constraint $0 \leq q_{(i,j)} < 1$ for all (i,j) and $q_{(1,1)} = 1 - (q_{(0,0)} + q_{(0,1)} + q_{(1,0)})$. The desirable characteristic of cost function $J(q)$ is that the parameters that achieve $J(q) = 0$ have the desired $\bar{v}_d$, $\bar{\theta}_d$ and $\bar{\sigma}_{\theta,d}^2$. Weighting coefficients in (16) will not change the location of minimum cost, and are therefore not used. Patches in other quadrants can be obtained by rotation.

5 Adaptive filtering with thin and fat patches

5.1 Thin and fat transition patches

When tracking a target, the more certain we are of target parameters, such as speed and heading, the lower we can

make the corresponding speed and heading variances, and vice versa (note that we only have 3 design variables $\bar{v}_d$, $\bar{\theta}_d$, and $\bar{\sigma}^2_{\theta,d}$).

The basic premise considered in this paper is that the performance of the HMM filter can be improved by designing the transition patch corresponding to dynamic properties matching those of the target but also with due consideration of the certainty about this knowledge. The transition patch design is achieved through equations (13)–(16). For example we might design a patch corresponding to known speed and heading with small heading variance, $\bar{\sigma}^2_\theta$. We will refer to patches with a small variance as "thin" patches and denote them q^T.

Unfortunately, in many practical situations there is a lack of *a priori* information about the target dynamics. For example, in the airborne collision avoidance application there is no clear way to anticipate the bearing at which the intruder aircraft may be approaching. If a thin patch is employed for filtering purposes here, the consequence is a heightened risk of gross mismatch between the patch dynamic properties and target dynamics, potentially leading to severely degraded filter performance.

In situations where there is initially a large degree of uncertainty about target dynamics, it may be more appropriate to deploy patches with a higher heading variance; that is "fat" patches, denoted q^F. This reduces the risk of gross mismatch in dynamic properties, but at the cost of slightly reduced filtering performance. Next, we propose an adaptive filtering approach, so that variation in model certainty can be handled, which adaptively switches between thin and fat patches. We will also exploit filter bank concepts [Lai *et al.*, 2008a].

5.2 An adaptive HMM filter bank

In this section, we propose an adaptive HMM filter bank, in which patch selection changes as we learn more about the dynamics. The filter bank has two modes of operation: target acquisition and target tracking.

Target acquisition

Initially the filter will be in target acquisition mode, with four fat filters — one corresponding to each quadrant of target heading. These filters are designed to allow detection of targets that might be moving in any direction. Four filters is a natural choice considering the two-dimensional nature of the images, but other numbers of filters could be used.

Target tracking

In the target tracking mode, we assume that the quadrant of target heading is known. We use four thin filters spaced equally across one quadrant.

Switching between modes

The filter bank starts initially in target acquisition mode (with four fat filters). When the detection metric, τ_k, of one of the filters crosses a switch threshold, η, we assume that there is a target with a heading in the quadrant corresponding to this filter. The filter bank then changes to tracking mode as follows:

1. The four fat filters, which are spread across 360°, are replaced by four thin filters, which are spread across the quadrant of target heading.
2. The state estimate, $\hat{X}_k$, of each of the four thin filters is initialised to that of the fat filter which detected the target.
3. The detection metric, τ_k of each of the four thin filters is initialised to that of the fat filter which detected the target.

To summarise this, let $\hat{X}_{s,k},\ s = 1, 2, 3, 4$ denote the sth filter in the filter bank, and let $q_{s,k}$ denote the transition patch used by the sth filter at time k. The adaptive filter operates as follows:

$$q_{s,k} = \begin{cases} q_s^F & \text{until } \max_s(\tau_{s,k}) > \eta \\ q_s^T & \text{afterwards,} \end{cases} \tag{17}$$

for each $s \in \{1, 2, 3, 4\}$, where q_s^F and q_s^T are the particular fat and thin patches respectively, chosen for the sth filter in the filter bank, $\tau_{s,k}$ is the detection metric for the sth filter at time k, and η is the switch threshold.

Remark 1 *When the filter is operating in the target tracking mode, the target heading can be inferred from which filter has the highest τ_k.*

Remark 2 *This preliminary approach does not handle events such as the losing track of the target (switching back to target acquisition mode), or detecting multiple targets. This sort of functionality is the topic of future research.*

Remark 3 *The choice of switch threshold, η, is currently an ad hoc decision based on trial and error.*

Remark 4 *For simplification of presentation we assume the target is well within a quadrant. In practical cases we could include filters from other quadrants rather than only using the four thin filters described above.*

6 Implementation

Hidden Markov model filtering of image sequences is a computationally intensive task. In this problem, the interaction between each pixel with all other pixels (i.e. the transition probabilities) must be considered, leading to N^2 relationships.

Equation (8) can be written in terms of each element of $\hat{X}_k$:

$$\hat{X}_{k,i} = N_k B(y_k)_{i,i} \sum_{j=1}^{N} A_{i,j} \hat{X}_{k-1,j}. \tag{18}$$

This equation needs to be evaluated for $i = 1, 2, ..., N$ (i.e. N times) at each time step. Rather than performing these calculations sequentially on traditional Central Processing Units (CPUs), we can exploit GPUs to perform many of the N calculations simultaneously, resulting in significantly reduced processing times. We developed a program that implemented (18) using CUDA, NVIDIA's parallel computing architecture for GPUs. For example, if a 512×512 resolution image is used, the algorithm uses 262144 threads. We note that the summation in (18) is only done over non-zero elements of $A_{i,j}$ to improve efficiency.

The program was run on a desktop machine with an NVIDIA Quadro 600 GPU, which has 96 cores and 1024MB RAM.

As a benchmark, a filter bank of four HMM filters with morphological preprocessing was run on a sequence of images of 512×512 resolution. The desktop machine processed these images at approximately 19 frames per second (FPS).

We note that the code used for this test was not highly optimised for fast processing, and significant increases in the frame rate could be achieved with further refinement. As an example of possible performance, a similar algorithm implemented in CUDA has been run in real-time, processing 1024×768 resolution images at 130 FPS [Lai *et al.*, 2010].

7 SIMULATION STUDIES AND REAL DATA EXPERIMENTS

In this section we investigate the properties and performance of the thin and fat filters, and investigate their use in an adaptive HMM filter. Simulations studies are described in Section 7.1, and the adaptive filter presented in Section 5 is tested with real data in Section 7.2.

7.1 Simulation studies

This section describes the simulations conducted to investigate the performance of the fat and thin filters. Firstly, our simulation environment is defined. Following this, the performance of fat and thin filters is compared with a parametric study of target heading and a receiver operating characteristic (ROC) curve. Finally, the adaptive filter presented in Section 5 is tested with a simulation.

Simulation set-up

These simulation studies aim to assess filter performance in environments similar to those found in airborne sense-and-avoid situations. The simulation set up is described here.

Background clutter Spatially correlated background clutter was created using a Gauss-Markov random field (GMRF)[Moura and Balram, 1992] with a resolution of 192×192 pixels, horizontal and vertical field interactions of 0.24 and driven by Gaussian noise $N(127, 333)$ of mean 127 and variance of 333. The noise values are discretised to whole numbers and limited to the range $0 - 255$.

Target A constant motion, constant intensity, extended target [Bell, 1993] of 3×3 pixels was added to the background clutter. The target's position was modelled as

$$\begin{bmatrix} p^x_{k+1} \\ p^y_{k+1} \end{bmatrix} = \begin{bmatrix} p^x_k \\ p^y_k \end{bmatrix} + \begin{bmatrix} v^x \\ v^y \end{bmatrix} \Delta T, \tag{19}$$

where $[p^x_k, p^y_k]$ is the target location in the image, given in pixel coordinates at time k and ΔT is the sampling period. The target position is discretised to the nearest pixel. The target intensity, I, is then added to the noise values in each pixel of the 3×3 neighbourhood to represent a target with extended area (size 3×3 pixels). The expected motion can be described in polar coordinates:

$$v = \sqrt{(v^x)^2 + (v^y)^2} \tag{20}$$

$$\theta = \arctan(v^y / v^x) \tag{21}$$

Units of speed are in pixels per frame.

Performance metrics In these simulations the primary method of comparison is the value of the detection metric, τ_k. Each test runs the filter over a sequence of 400 frames, and the mean value of τ_k over the last 300 frames is used as the metric of performance. This value is referred to as the mean steady-state τ_k. Because τ_k is related to confidence in target presence, a higher τ_k for a given target sequence is equivalent to increasing the SNR of the filter output.

We now consider studies supporting use of thin patches when target dynamics certainty is high.

Parametric study of target heading

A parametric study of target heading was conducted in which the performance of five transition patches is examined for different target headings. The image sequences used had a target of speed $v = 0.5$, intensity of $I = 21$, moving at headings between 0 and $\frac{\pi}{2}$. Four of the filters used thin patches with heading variance $\bar{\sigma}^2_\theta = 0.26$, expected speed $\bar{v} = 0.5$, and expected headings, μ_θ, distributed equally across the first quadrant ($\frac{\pi}{16}$, $\frac{3\pi}{16}$, $\frac{5\pi}{16}$ and $\frac{7\pi}{16}$). The fifth filter had a fat patch with heading variance $\bar{\sigma}^2_\theta = 0.6$, expected heading $\bar{\theta} = \frac{\pi}{4}$ and expected speed $\bar{v} = 0.5$. Additionally, a patch with equal probabilities in the target quadrant and zeros elsewhere (as used in [Lai *et al.*, 2008a]), denoted q^L was used. A Monte Carlo simulation was conducted with 40 trials, and the output of this is shown in Figure 2.

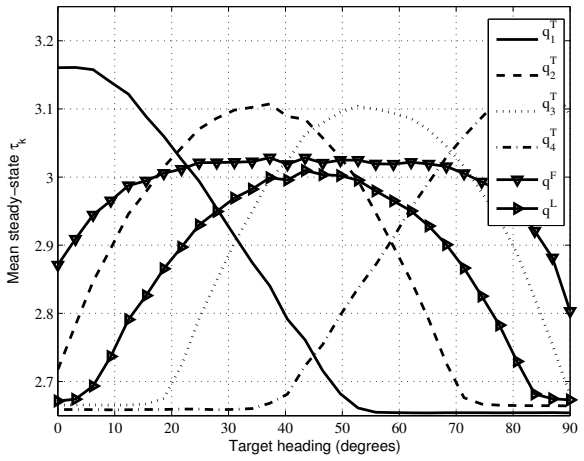

Figure 2: A parametric study of target heading for four thin filters and one fat filter. The four thin filters in combination cover the entire quadrant.

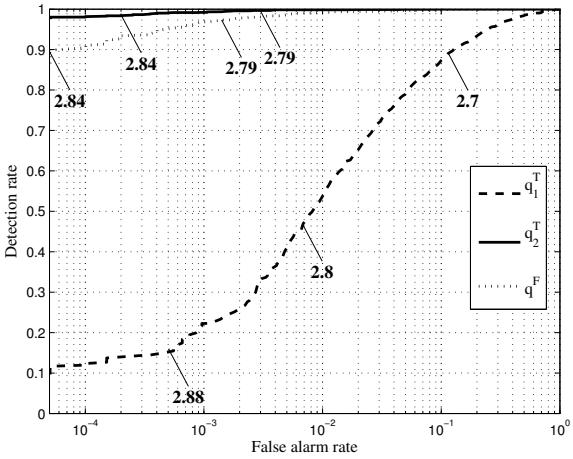

Figure 3: Receiver operating characteristic curves for fat and thin filters. The solid line represents the filter with patch q_2^T, (a thin patch matching the true dynamics). Also shown is a thin patch which is mismatched to the true dynamics, q_1^T, and a fat patch in the correct quadrant q^F. The numbers on the plot represent the detection threshold at that location on the curve.

Figure 2 shows that for all headings in the range considered, the best of the collection of thin filters, is better on average than the fat filter. We note that the thin filters do not all have the same peak magnitude, which is an indication that the best way to cover target headings between 0 and $\frac{\pi}{2}$ is not necessarily through thin filters equally distributed within the quadrant. The strength of τ_k is not necessarily the only way to understand performance of these filters, and in the next section we consider detection versus false alarm performance with a ROC curve. However, the above study does give an indication of filter fatness.

Receiver operating characteristic curve

In this study, the false alarm/detection rate performance of the thin and fat patches are compared with a ROC curve. Three filters were used with the following patches, respectively:

- q_1^T: a thin patch not matching true dynamics;
- q_2^T: a thin patch matching true dynamics; and
- q^F: a fat patch in the quadrant of the true target heading.

To calculate the false alarm rate, a sequence of 20000 synthetic images with no target was run through the filters. For a given detection threshold, γ, the false alarm rate is calculated as the number of times $\tau_k > \gamma$, divided by the number of frames (first 200 frames used for initialisation and excluded from count). For the detection rate, a sequence of 250 synthetic images with a target with intensity of $I = 18$ was run through the filters. If at the end of the sequence, the detection statistic, τ_k, was greater than the threshold and the estimated position of the target was within 10 pixels of the true position, a detection was declared. The detection simulation was repeated for 1900 Monte Carlo trials.

The calculated ROC in Figure 3 shows the detection rate versus the false alarm rate for a range of different thresholds (γ). Figure 3 shows that, for this data, the thin patch that matches the true dynamics has better false alarm/detection performance than the fat patch, but the fat patch performs better than a mismatched thin patch.

We will now consider the performance of a combination of fat and thin filters in an adaptive HMM filter.

Study: switching from fat to thin patches

In this section the adaptive HMM filter proposed in Section 5 is examined. Synthetic data is used, with a target moving at $v = 0.5$ pixels per frame at a heading of $\theta = \frac{3\pi}{16}$. An example frame is shown in Figure 4. The adaptive HMM filter starts with four fat filters with variance $\bar{\sigma}_\theta^2 = 0.6$, expected speed, $\bar{v} = 0.5$ pixels per frame, and expected headings distributed around 360°, $(\frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4})$. After the switch, the adaptive HMM filter has four thin filters with variance $\bar{\sigma}_\theta^2 = 0.26$, expected speed $\bar{v} = 0.5$, and expected headings distributed around the quadrant of target heading, e.g. $(\frac{\pi}{16}, \frac{3\pi}{16}, \frac{5\pi}{16}, \frac{7\pi}{16})$ if the target is in the first quadrant.

For comparison, we show the performance of the filter when it is not allowed to switch, i.e. a bank of four fat filters for the whole test.

The plot in Figure 5 shows the performance of the

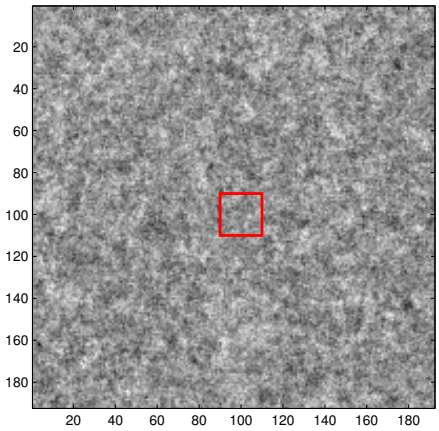

Figure 4: An example synthetic data image frame with spatial correlated noise. The target is located in the centre of the square, but is difficult to see amongst the clutter.

adaptive and non-adaptive filters over time. The two lines represents the highest τ_k from each filter bank. The target enters the image at frame 150, and the adaptive filter switches at frame 227. Figure 5 demonstrates that τ_k is higher in the adaptive filter after the switch.

7.2 Real data

In this section the adaptive HMM filter proposed in Section 5 is tested using real data.

Data collection

A flight test was conducted to collect airborne images from realistic collision course scenarios. A custom modified Cessna 172 [Greer *et al.*, 2010] was used as the camera platform, and was flown towards a Cessna 182 in a head-on geometry (with an altitude separation for safety).

A camera mounted on the wing strut of the Cessna 172 recorded visual spectrum, grey scale images at a rate of 15Hz, with a field of view of approximately 20°×15°. The images have a bit depth of 8 bits per pixel and a resolution of 1024×768 pixels. Aircraft state data was recorded with the images and allowed ego-motion compensation to be performed. The weather during the flight was clear, and there was no cloud in the images used for this test. This data collection methodology was also used in [Lai *et al.*, 2011].

The images were cropped to 256×256 in order to reduce processing time for this experiment. An example frame from this sequence is shown in Figure 6, with the target highlighted.

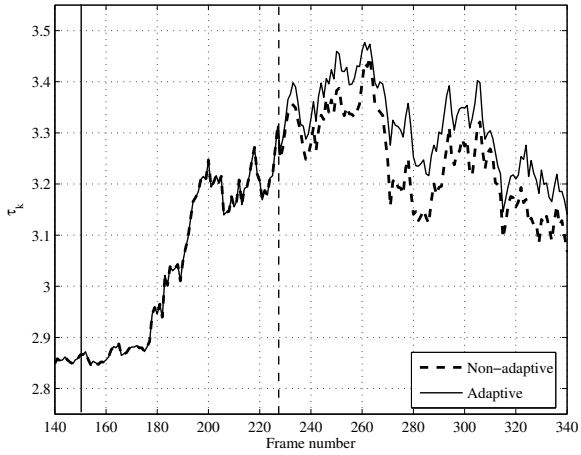

Figure 5: Detection statistics over time for two filters. The solid line represents the adaptive filter bank, and the dashed line represents a bank of fat filters. The target enters the image at frame 150 (marked by the solid vertical line), and the filter switches at frame 227 (marked by the dashed vertical line).

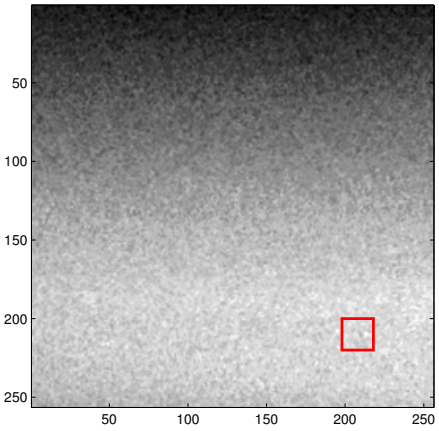

Figure 6: An example frame from the real data sequence. The target is highlighted by the box.

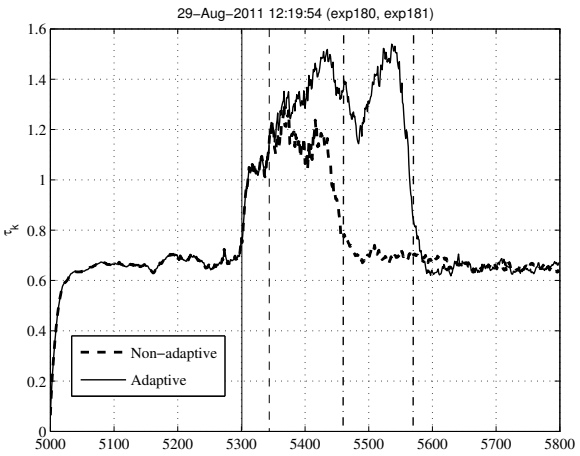

Figure 7: Detection statistics over time for an adaptive HMM filter running on a real data sequence. The target enters the image at approximately frame 5300 (marked by the solid vertical line), and the filter switches at frame 5344 (marked by the dashed vertical line). The non-adaptive filter loses track of the target at approximately frame 5460 (marked by the first dot-dashed vertical line), and the adaptive filter loses track of the target at approximately frame 5570 (marked by the second dot-dashed vertical line). The target leaves the image at about frame 5630.

Study: switching from fat to thin with real data The HMM filter proposed in Section 5 was run on the data collected in the flight test described above. The test configuration is the same as the adaptive filter test in Section 7.1; i.e. we compare the performance of an adaptive filer to a non-adaptive filter without switching capability. In this test the target enters at approximately frame 5300 and leaves at approximately frame 5630. The adaptive filter switches at frame 5344. Figure 7 demonstrates that τ_k is higher for the adaptive filter after the switch. Both filters lost track of the target before it leaves the field of view. The τ_k values start to decrease as the filters lose track of the target. As shown in Figure 7, the adaptive filter tracks the target for longer than the non-adaptive filter.

7.3 Summary

This section has investigated the effect of filter "fatness" on detection performance. Simulations showed that a well matched thin filter performs better than a fat filter, and that a fat filter performs better than a poorly matched thin filter. Building from this, it was shown that, under these conditions, a bank of thin filters will perform better than a fat filter. These two observations justify the use of fat filters when there is uncertainty about target heading, and thin filters once the heading has been determined — supporting the idea of using an adaptive filter. An adaptive filter based on these principles was shown to be effective with synthetic and real data.

8 CONCLUSION

This paper illustrates the benefits of adapting the fatness of filter design to the certainty of target motion in a track-before-detect approach to target tracking in image sequences.

9 Acknowledgements

Computational resources and services used in this work were provided by the HPC and Research Support Unit, Queensland University of Technology, Brisbane, Australia.

This research was partially supported under the Australian Research Councils Linkage Projects Scheme (project number LP100100302) and the Smart Skies Project, which is funded, in part, by the Queensland State Government Smart State Funding Scheme.

The authors would also like to thank those involved in the airborne data collection: Scott McNamara for providing engineering support Reece Clothier for organisational support, and the pilots, Rhys Mudford and Duncan Greer.

References

[Barniv and Kella, 1987] Y. Barniv and O. Kella. Dynamic programming solution for detecting dim moving targets part ii: Analysis. *Aerospace and Electronic Systems, IEEE Transactions on*, AES-23(6):776 –788, nov. 1987.

[Barniv, 1985] Y. Barniv. Dynamic programming solution for detecting dim moving targets. *IEEE Transactions on Aerospace and Electronic Systems*, AES-21:144–156, 1985.

[Bell, 1993] M.R. Bell. Information theory and radar waveform design. *Information Theory, IEEE Transactions on*, 39(5):1578 –1597, September 1993.

[Bruno and Moura, 2001] M.G.S. Bruno and J.M.F. Moura. Multiframe detector/tracker: optimal performance. *Aerospace and Electronic Systems, IEEE Transactions on*, 37(3):925 –945, jul. 2001.

[Bruno, 2004] M.G.S. Bruno. Bayesian methods for multiaspect target tracking in image sequences. *Signal Processing, IEEE Transactions on*, 52(7):1848 – 1861, jul. 2004.

[Carnie *et al.*, 2006] R. Carnie, R. Walker, and P. Corke. Image processing algorithms for UAV "sense and

avoid". In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Orlando, FL, 2006.

[Davey *et al.*, 2008] S. J. Davey, M. G. Rutten, and B. Cheung. A comparison of detection performance for several track-before-detect algorithms. *EURASIP Journal on Advances in Signal Processing*, 2008:1–10, 2008.

[DeGarmo, 2004] M. T. DeGarmo. Issues concerning integration of unmanned aerial vehicles in civil airspace. Technical report, The MITRE Corporation, 2004.

[Dougherty and Lotufo, 2003] E.R. Dougherty and R.A. Lotufo. *Hands-on morphological image processing*. Tutorial texts in optical engineering. SPIE Press, 2003.

[Gandhi *et al.*, 2006] T. Gandhi, M.-T. Yang, R. Kasturi, O. Camps, L. Coraor, and J. McCandless. Performance characterisation of the dynamic programming obstacle detection algorithm. *IEEE Transactions on Image Processing*, 15:1202–1214, 2006.

[Geyer *et al.*, 2009] C. Geyer, D. Dey, and S. Singh. Prototype sense-and-avoid system for UAVs. Technical report, Robotics Institute, Carnegie Mellon University, 2009. CMU-RI-TR-09-09.

[Gonzalez *et al.*, 2004] R. C. Gonzalez, R. E. Woods, and S. L. Eddins. *Digital Image Processing using MATLAB*. Upper Saddle River, NJ: Pearson Prentice Hall, 2004.

[Greer *et al.*, 2010] Duncan G. Greer, Rhys Mudford, Damien Dusha, and Rodney A. Walker. Airborne systems laboratory for automation research. In *27th International Congress of the Aeronautical Sciences*, Acropolis Conference Centre, Nice, 2010. ICAS.

[Lai and Ford, 2010] J. Lai and J.J. Ford. Relative entropy rate based multiple hidden Markov model approximation. *Signal Processing, IEEE Transactions on*, 58(1):165 –174, jan. 2010.

[Lai *et al.*, 2008a] J. Lai, J. J. Ford, P. O'Shea, and R. Walker. Hidden Markov model filter banks for dim target detection from image sequences. In *Proceedings of Digital Image Computing: Techniques and Applications (DICTA)*, Canberra, 2008.

[Lai *et al.*, 2008b] J. Lai, J. J. Ford, P. O'Shea, R. Walker, and M. Bosse. A study of morphological pre-processing approaches for track-before-detect dim target detection. In *Proceedings of Australasian Conference on Robotics and Automation (ACRA)*, Canberra, 2008.

[Lai *et al.*, 2010] J. Lai, L Mejias, and J.J. Ford. Airborne vision-based collision-detection system. *Journal of Field Robotics*, 28(2), 2010.

[Lai *et al.*, 2011] J. Lai, J. J. Ford, L. Mejias, P. O'Shea, and R. Walker. Detection versus false alarm characterisation of a vision-based airborne dim-target collision detection system. In *Submitted to Proceedings of Digital Image Computing: Techniques and Applications (DICTA)*, 2011. in press.

[Limitations of the See-and-Avoid Principle, 1991] Limitations of the see-and-avoid principle. Technical report, Australian Transport Safety Bureau, Australian Government, 1991.

[Moura and Balram, 1992] J.M.F. Moura and N. Balram. Recursive structure of noncausal gauss-markov random fields. *Information Theory, IEEE Transactions on*, 38(2):334 –354, March 1992.

[Tonissen and Evans, 1996] S. M. Tonissen and R. J. Evans. Performance of dynamic programming techniques for track-before-detect. *IEEE Transactions on Aerospace and Electronic Systems*, 32:1440–1451, 1996.